Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	16227	(file adj3 system) or file?system	USPAT	OR	OFF	2005/12/27 15:13
S2	491696	log$4 or history	USPAT	OR	OFF	2005/12/27 10:51
S3	95344	undo or redo or rollback or restore or reconstruct	USPAT	OR	OFF	2005/12/23 09:31
S4	16227	(file adj3 system) or file?system	USPAT	OR	OFF	2005/12/23 09:31
S5	491696	log$4 or history	USPAT	OR	OFF	2005/12/23 09:31
S6	2808	S4 and S5 and S3	USPAT	OR	OFF	2005/12/23 09:07
S7	128	S4 same S5 same S3	USPAT	OR	OFF	2005/12/23 09:31
S8	31	S4 with S5 with S3	USPAT	OR	OFF	2005/12/23 09:07
S9	175022	synchroniz$4	USPAT	OR	ON	2005/12/27 09:48
S10	19908	(file adj3 system) or file?system	USPAT	OR	ON	2005/12/23 09:31
S11	493046	log$4 or history	USPAT	OR	ON	2005/12/23 09:31
S12	113182	undo or redo or rollback or restore or reconstruct	USPAT	OR	ON	2005/12/23 09:31
S13	2	S9 same S10 same S11 same S12	USPAT	OR	ON	2005/12/23 09:33
S14	12	S9 same S10 same S12	USPAT	OR	ON	2005/12/23 09:41
S15	392	S9 same S10	USPAT	OR	ON	2005/12/23 09:41
S16	187	S9 with S10	USPAT	OR	ON	2005/12/23 09:41
S17	182	S9 with S10 not S14	USPAT	OR	ON	2005/12/23 09:41
S18	15	("5835953" | "5907672" | "5996086" | "6000039" | "6078932" | "6119244" | "6131088" | "6212531" | "6360330" | "6377951" | "6453325" | "6529921" | "6529944" | "6643671" | "6668264").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/12/23 10:19
S19	179509	synchroniz$4 or reconcil$4 or undoable	USPAT	OR	ON	2005/12/27 09:50
S20	175260	synchroniz$4	USPAT	OR	ON	2005/12/27 09:50
S21	19964	(file adj3 system) or file?system	USPAT	OR	ON	2005/12/27 09:50
S22	113336	undo or redo or rollback or restore or reconstruct	USPAT	OR	ON	2005/12/27 09:50
S23	12	S20 same S21 same S22	USPAT	OR	ON	2005/12/27 09:50
S24	182	S20 with S21 not S23	USPAT	OR	ON	2005/12/27 09:50
S25	4189	S21 and S19 not S24	USPAT	OR	ON	2005/12/27 09:50
S26	20	S21 with S19 not S24	USPAT	OR	ON	2005/12/27 09:55
S27	250	S21 same S19 not S24	USPAT	OR	ON	2005/12/27 11:27
S28	68239	"707".clas. or "713".clas. or "714". clas. or "715".clas. "717".clas.	USPAT	OR	OFF	2005/12/27 09:58
S29	179509	synchroniz$4 or reconcil$4 or undoable	USPAT	OR	ON	2005/12/27 09:56

S30	175260	synchroniz$4	USPAT	OR	ON	2005/12/27 09:56
S31	19964	(file adj3 system) or file?system	USPAT	OR	ON	2005/12/27 11:12
S32	113336	undo or redo or rollback or restore or reconstruct	USPAT	OR	ON	2005/12/27 09:56
S33	12	S30 same S31 same S32	USPAT	OR	ON	2005/12/27 09:56
S34	182	S30 with S31 not S33	USPAT	OR	ON	2005/12/27 09:56
S35	250	S31 same S29 not S34	USPAT	OR	ON	2005/12/27 09:56
S36	152	S35 and S28	USPAT	OR	OFF	2005/12/27 09:56
S37	41296	"707".clas. or "714".clas.	USPAT	OR	OFF	2005/12/27 10:48
S38	125	S35 and S37	USPAT	OR	OFF	2005/12/27 09:59
S39	250	S31 same S29 not S34	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/12/27 11:10
S40	57198	"707".clas. or "709".clas. "714".clas.	USPAT	OR	OFF	2005/12/27 10:48
S44	212	S31 with S32 and S40	USPAT	OR	OFF	2005/12/27 10:52
S45	449	S31 same S32 and S40	USPAT	OR	OFF	2005/12/27 10:49
S46	419	S31 same S32 and S40 not S39	USPAT	OR	OFF	2005/12/27 11:10
S48	521819	log$4 or history or schedule	USPAT	OR	OFF	2005/12/27 12:56
S49	1862	S31 and S32 and S48 and S40	USPAT	OR	OFF	2005/12/27 10:52
S50	146	S31 same S32 same S48 and S40	USPAT	OR	OFF	2005/12/27 10:52
S51	64	S31 with S32 same S48 and S40	USPAT	OR	OFF	2005/12/27 10:52
S52	45	S31 with S32 with S48 and S40	USPAT	OR	OFF	2005/12/27 10:52
S53	302	share$4 adj2 ((file adj3 system) or file?system)	USPAT	OR	ON	2005/12/27 11:13
S54	112	S53 and S29	USPAT	OR	ON	2005/12/27 11:14
S55	98	S53 and S29 and S48	USPAT	OR	ON	2005/12/27 11:14
S56	26	S53 same S29 and S48	USPAT	OR	ON	2005/12/27 11:15
S57	1	S53 same S29 same S48	USPAT	OR	ON	2005/12/27 11:14
S58	26	S53 same S29 and S48	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/12/27 11:18
S59	20	S53 with S29	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/12/27 11:18

S60	2	S53 with S29 not S58	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/12/27 11:19
S61	2	S53 same S29 not S58	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/12/27 11:19
S62	28	S53 same S29	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/12/27 11:19
S63	432	S31 same S29	USPAT	OR	ON	2005/12/27 11:27
S64	377	S31 same S29 AND S48	USPAT	OR	ON	2005/12/27 11:28
S65	74	S31 same S29 same S48	USPAT	OR	ON	2005/12/27 12:57
S66	522763	log$4 or history or schedule or audit	USPAT	OR	OFF	2005/12/27 12:57
S67	179509	synchroniz$4 or reconcil$4 or undoable	USPAT	OR	ON	2005/12/27 12:57
S68	19964	(file adj3 system) or file?system	USPAT	OR	ON	2005/12/27 12:57
S69	74	S68 same S67 same S66	USPAT	OR	ON	2005/12/27 14:21
S70	10	("5745750").URPN.	USPAT	OR	OFF	2005/12/27 13:00
S71	0	("6925476").URPN.	USPAT	OR	OFF	2005/12/27 13:24

S72	94	("20010044805" | "20020138765" | "20030037020" | "20030069874" | "5130993" | "5392390" | "5519606" | "5623661" | "5628005" | "5630081" | "5638508" | "5649195" | "5666553" | "5682524" | "5684990" | "5694596" | "5701423" | "5706509" | "5710922" | "5727202" | "5729735" | "5729743" | "5742792" | "5745750" | "5745906" | "5758150" | "5768597" | "5771354" | "5778346" | "5778388" | "5787247" | "5787262" | "5809497" | "5812773" | "5812793" | "5832489" | "5832518" | "5832519" | "5845283" | "5875296" | "5884323" | "5884325" | "5893119" | "5897640" | "5897642" | "5935262" | "5937405" | "5943676" | "5961590" | "5968131" | "5974238" | "5974563" | "6000000" | "6006274" | "6012063" | "6016478" | "6023708" | "6023723" | "6034621" | "6044381" | "6052735" | "6058399" | "6061790" | "6065018" | "6081900" | "6131096" | "6131116" | "6141011" | "6141664" | "6151606" | "6182117" | "6202085" | "6205448" | "6212529" | "6216131" | "6219694" | "6223187" | "6226650" | "6247135" | "6272545" | "6275831" | "6282698" | "6295541" | "6304881" | "6324544" | "6330568" | "6397351" | "6401104" | "6405218" | "6449622" | "6457062" | "6487560" | "6684206" | "6718348").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/12/27 13:25
S73	1	"20030069874".PN.	US-PGPUB	OR	OFF	2005/12/27 13:25
S74	10	("5745750").URPN.	USPAT	OR	OFF	2005/12/27 14:15
S75	1	"5857204".pn.	USPAT	OR	OFF	2005/12/27 15:13
S76	905	((file adj3 system) or file?system) and primitive	USPAT	OR	OFF	2005/12/27 15:14
S77	55	((file adj3 system) or file?system) with primitive	USPAT	OR	OFF	2005/12/27 15:22
S78	195	((file adj3 system) or file?system) same (primitive or atomic)	USPAT	OR	OFF	2005/12/27 15:22
S79	522763	log$4 or history or schedule or audit	USPAT	OR	OFF	2005/12/27 15:23
S80	24	((file adj3 system) or file?system) same (primitive or atomic) same S79	USPAT	OR	OFF	2005/12/27 15:48

S81	179509	synchroniz$4 or reconcil$4 or undoable	USPAT	OR	ON	2005/12/27 15:54
S82	2816	(primitive or atomic adj action or basic adj action) and S79 and S81	USPAT	OR	ON	2005/12/27 15:54
S83	16955	(primitive or atomic adj action or basic adj action)	USPAT	OR	ON	2005/12/27 15:54
S84	2816	S83 and S79 and S81	USPAT	OR	ON	2005/12/27 15:54
S85	55	S83 same S79 same S81	USPAT	OR	ON	2005/12/27 15:56
S86	57198	"707".clas. or "709".clas. "714".clas.	USPAT	OR	OFF	2005/12/27 15:56
S87	14	S83 same S79 same S81 and S86	USPAT	OR	ON	2005/12/27 15:56

Terms used **file system log synchronize undoable** Found **8,296** of **169,166**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 A Survey of Techniques for Synchronization and Recovery in Decentralized Computer Systems

Walter H. Kohler

June 1981 **ACM Computing Surveys (CSUR)**, Volume 13 Issue 2

Publisher: ACM Press

Full text available: pdf(3.33 MB) Additional Information: full citation, references, citings, index terms

2 Speculative execution in a distributed file system

Edmund B. Nightingale, Peter M. Chen, Jason Flinn

October 2005 **ACM SIGOPS Operating Systems Review , Proceedings of the twentieth ACM symposium on Operating systems principles SOSP '05**, Volume 39 Issue 5

Publisher: ACM Press

Full text available: pdf(305.54 KB) Additional Information: full citation, abstract, references, index terms

Speculator provides Linux kernel support for speculative execution. It allows multiple processes to share speculative state by tracking causal dependencies propagated through inter-process communication. It guarantees correct execution by preventing speculative processes from externalizing output, e.g., sending a network message or writing to the screen, until the speculations on which that output depends have proven to be correct. Speculator improves the performance of distributed file systems ...

Keywords: causality, distributed file systems, speculative execution

3 File servers for network-based distributed systems

Liba Svobodova

December 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 4

Publisher: ACM Press

Full text available: pdf(4.23 MB) Additional Information: full citation, references, citings, index terms, review

4 Synchronization and recovery in a client-server storage system

E. Panagos, A. Biliris

August 1997 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 6 Issue 3

Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(205.25 KB) Additional Information: full citation, abstract, citings, index terms

Client-server object-oriented database management systems differ significantly from traditional centralized systems in terms of their architecture and the applications they

target. In this paper, we present the client-server architecture of the EOS storage manager and we describe the concurrency control and recovery mechanisms it employs. EOS offers a semi-optimistic locking scheme based on the multi-granularity two-version two-phase locking protocol. Under this scheme, multiple concurrent reade ...

Keywords: Checkpoint, Client-server architecture, Object management, Concurrency control, Locking, Logging, Recovery, Transaction management

5 Transactions and synchronization in a distributed operating system

Matthew J. Weinstein, Thomas W. Page, Brian K. Livezey, Gerald J. Popek

December 1985 **ACM SIGOPS Operating Systems Review , Proceedings of the tenth ACM symposium on Operating systems principles SOSP '85**, Volume 19 Issue 5

Publisher: ACM Press

Full text available: pdf(974.32 KB) Additional Information: full citation, references, citings, index terms

6 Synchronization and recovery of actions

J. E. Allchin, M. S. McKendry

January 1985 **ACM SIGOPS Operating Systems Review**, Volume 19 Issue 1

Publisher: ACM Press

Full text available: pdf(1.26 MB) Additional Information: full citation, abstract, references

We introduce an approach to robust computation in distributed systems. This approach is the foundation for reliability in the **Clouds** decentralized operating system. It is based on atomic actions operating on instances of abstract data types (objects). We present an event-based model of computation in which scheduling of responses to operation invocations is controlled by objects. We discuss an integrated strategy for synchronization *and* recovery which uses relationships between the ...

7 Synchronization and recovery of actions

J. E. Allchin, M. S. McKendry

August 1983 **Proceedings of the second annual ACM symposium on Principles of distributed computing**

Publisher: ACM Press

Full text available: pdf(1.29 MB) Additional Information: full citation, abstract, references, citings, index terms

We introduce an approach to robust computation in distributed systems. This approach is the foundation for reliability in the Clouds decentralized operating system. It is based on atomic actions operating on instances of abstract data types (objects). We present an event-based model of computation in which scheduling of responses to operation invocations is controlled by objects. We discuss an integrated strategy for synchronization and recovery which uses rela ...

8 Managing update conflicts in Bayou, a weakly connected replicated storage system

D. B. Terry, M. M. Theimer, Karin Petersen, A. J. Demers, M. J. Spreitzer, C. H. Hauser

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5

Publisher: ACM Press

Full text available: pdf(1.56 MB) Additional Information: full citation, references, citings, index terms

9 Replication in the harp file system

Barbara Liskov, Sanjay Ghemawat, Robert Gruber, Paul Johnson, Liuba Shrira

September 1991 **ACM SIGOPS Operating Systems Review , Proceedings of the thirteenth ACM symposium on Operating systems principles SOSP '91**, Volume 25 Issue 5

Publisher: ACM Press

Additional Information: full citation, abstract, references, citings, index

Full text available: pdf(1.60 MB) terms

This paper describes the design and implementation of the Harp file system. Harp is a replicated Unix file system accessible via the VFS interface. It provides highly available and reliable storage for files and guarantees that file operations are executed atomically in spite of concurrency and failures. It uses a novel variation of the primary copy replication technique that provides good performance because it allows us to trade disk accesses for network communication. Harp is intended to be u ...

10 The Alpine file system

M. R. Brown, K. N. Kolling, E. A. Taft
November 1985 **ACM Transactions on Computer Systems (TOCS)**, Volume 3 Issue 4
Publisher: ACM Press

Full text available: pdf(2.95 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Alpine is a file system that supports atomic transactions and is designed to operate as a service on a computer network. Alpine's primary purpose is to store files that represent databases. An important secondary goal is to store ordinary files representing documents, program modules, and the like. Unlike other file servers described in the literature, Alpine uses a log-based technique to implement atomic file update. Another unusual aspect of Alpine is that it performs all commu ...

11 Multi-level transaction management for complex objects: implementation, performance, parallelism

Gerhard Weikum, Christof Hasse
October 1993 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 2 Issue 4
Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(2.83 MB) Additional Information: full citation, abstract, references, citings

Multi-level transactions are a variant of open-nested transactions in which the subtransactions correspond to operations at different levels of a layered system architecture. They allow the exploitation of semantics of high-level operations to increase concurrency. As a consequence, undoing a transaction requires compensation of completed subtransactions. In addition, multi-level recovery methods must take into consideration that high-level operations are not necessarily atomic if multiple pages ...

Keywords: atomicity, complex objects, inter- and intratransaction parallelism, multi-level transactions, performance, persistence, recovery

12 Virtual memory management for database systems

Irving L. Traiger
October 1982 **ACM SIGOPS Operating Systems Review**, Volume 16 Issue 4
Publisher: ACM Press

Full text available: pdf(2.08 MB) Additional Information: full citation, abstract, references, citings

Over the last several years, a number of hardware and software systems have been developed which map entire files directly into the virtual memory address spaces used by programs. Since all file contents are directly addressable, there is no need for a programmer to issue explicit file system actions, such as Read or Write. In addition, all of the buffer management problems are eliminated, since programmers do not have to squeeze pieces of large files into small virtual spaces. Although these ad ...

13 Principles of transaction-oriented database recovery

Theo Haerder, Andreas Reuter
December 1983 **ACM Computing Surveys (CSUR)**, Volume 15 Issue 4
Publisher: ACM Press

Full text available: pdf(2.48 MB) Additional Information: full citation, references, citings, index terms, review

14 Distributed, object-based programming systems

Roger S. Chin, Samuel T. Chanson
March 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 1
Publisher: ACM Press

Full text available: pdf(2.97 MB) Additional Information: full citation, abstract, references, citings, index terms, review

The development of distributed operating systems and object-based programming languages makes possible an environment in which programs consisting of a set of interacting modules, or objects, may execute concurrently on a collection of loosely coupled processors. An object-based programming language encourages a methodology for designing and creating a program as a set of autonomous components, whereas a distributed operating system permits a collection of workstations or personal computers ...

Keywords: capability scheme, distributed operating systems, error recovery, method invocation, nested transaction, object model, object reliability, object-based programming languages, processor allocation, resource management, synchronization, transaction

15 User Recovery and Reversal in Interactive Systems

James E. Archer, Richard Conway, Fred B. Schneider
January 1984 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 6 Issue 1
Publisher: ACM Press
Full text available: pdf(1.30 MB) Additional Information: full citation, references, citings, index terms

16 ARIES: a transaction recovery method supporting fine-granularity locking and partial rollbacks using write-ahead logging

C. Mohan, Don Haderle, Bruce Lindsay, Hamid Pirahesh, Peter Schwarz
March 1992 **ACM Transactions on Database Systems (TODS)**, Volume 17 Issue 1
Publisher: ACM Press

Full text available: pdf(5.23 MB) Additional Information: full citation, abstract, references, citings, index terms, review

DB2TM, IMS, and TandemTM systems. ARIES is applicable not only to database management systems but also to persistent object-oriented languages, recoverable file systems and transaction-based operating systems. ARIES has been implemented, to varying degrees, in IBM's OS/2TM Extended Edition Database Manager, DB2, Workstation Data Save Facility/VM, Starburst and QuickSilver, and in the University of Wisconsin's EXODUS and Gamma d ...

Keywords: buffer management, latching, locking, space management, write-ahead logging

17 Anatomy of a native XML base management system

T. Fiebig, S. Helmer, C.-C. Kanne, G. Moerkotte, J. Neumann, R. Schiele, T. Westmann
December 2002 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 11 Issue 4
Publisher: Springer-Verlag New York, Inc.
Full text available: pdf(300.97 KB) Additional Information: full citation, abstract, citings, index terms

Several alternatives to manage large XML document collections exist, ranging from file systems over relational or other database systems to specifically tailored XML base management systems. In this paper we give a tour of Natix, a database management system designed from scratch for storing and processing XML data. Contrary to the common belief that management of XML data is just another application for traditional databases like relational systems, we illustrate how almost every component in a ...

Keywords: Database, XML

18 Experience with transactions in QuickSilver

Frank Schmuck, Jim Wylie
September 1991 **ACM SIGOPS Operating Systems Review , Proceedings of the thirteenth ACM symposium on Operating systems principles SOSP '91**, Volume 25 Issue 5

Publisher: ACM Press

Full text available: pdf(1.66 MB) Additional Information: full citation, abstract, references, citings, index terms

All programs in the QuickSilver distributed system behave atomically with respect to their updates to permanent data. Operating system support for *transactions* provides the framework required to support this, as well as a mechanism that unifies reclamation of resources after failures or normal process termination. This paper evaluates the use of transactions for these purposes in a general purpose operating system and presents some of the lessons learned from our experience with a complet ...

19 Report on the fourth ACM SIGOPS European workshop fault tolerance support in distributed systems

Özalp Babaoğlu
January 1991 **ACM SIGOPS Operating Systems Review**, Volume 25 Issue 1

Publisher: ACM Press

Full text available: pdf(1.76 MB) Additional Information: full citation, index terms

20 Extended ephemeral logging: log storage management for applications with long lived transactions

John S. Keen, William J. Dally
March 1997 **ACM Transactions on Database Systems (TODS)**, Volume 22 Issue 1

Publisher: ACM Press

Full text available: pdf(566.34 KB) Additional Information: full citation, references, index terms, review

Keywords: OLTP, disk management, logging, long transactions

Terms used
file system schedule primitive synchronize undoable

Found **2,492** of **169,166**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 A Survey of Techniques for Synchronization and Recovery in Decentralized Computer Systems

Walter H. Kohler

June 1981 **ACM Computing Surveys (CSUR)**, Volume 13 Issue 2

Publisher: ACM Press

Full text available: pdf(3.33 MB) Additional Information: full citation, references, citings, index terms

2 Synchronization and recovery of actions

J. E. Allchin, M. S. McKendry

August 1983 **Proceedings of the second annual ACM symposium on Principles of distributed computing**

Publisher: ACM Press

Full text available: pdf(1.29 MB) Additional Information: full citation, abstract, references, citings, index terms

We introduce an approach to robust computation in distributed systems. This approach is the foundation for reliability in the Clouds decentralized operating system. It is based on atomic actions operating on instances of abstract data types (objects). We present an event-based model of computation in which scheduling of responses to operation invocations is controlled by objects. We discuss an integrated strategy for synchronization and recovery which uses rela ...

3 Synchronization and recovery of actions

J. E. Allchin, M. S. McKendry

January 1985 **ACM SIGOPS Operating Systems Review**, Volume 19 Issue 1

Publisher: ACM Press

Full text available: pdf(1.26 MB) Additional Information: full citation, abstract, references

We introduce an approach to robust computation in distributed systems. This approach is the foundation for reliability in the **Clouds** decentralized operating system. It is based on atomic actions operating on instances of abstract data types (objects). We present an event-based model of computation in which scheduling of responses to operation invocations is controlled by objects. We discuss an integrated strategy for synchronization *and* recovery which uses relationships between the ...

4 Distributed systems - programming and management: On remote procedure call

Patrícia Gomes Soares

November 1992 **Proceedings of the 1992 conference of the Centre for Advanced Studies on Collaborative research - Volume 2**

Publisher: IBM Press

Full text available: pdf(4.52 MB) Additional Information: full citation, abstract, references, citings

The Remote Procedure Call (RPC) paradigm is reviewed. The concept is described, along with the backbone structure of the mechanisms that support it. An overview of works in supporting these mechanisms is discussed. Extensions to the paradigm that have been proposed to enlarge its suitability, are studied. The main contributions of this paper are a standard view and classification of RPC mechanisms according to different perspectives, and a snapshot of the paradigm in use today and of goals for t ...

5 Distributed, object-based programming systems

Roger S. Chin, Samuel T. Chanson
March 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 1
Publisher: ACM Press

Full text available: pdf(2.97 MB) Additional Information: full citation, abstract, references, citings, index terms, review

The development of distributed operating systems and object-based programming languages makes possible an environment in which programs consisting of a set of interacting modules, or objects, may execute concurrently on a collection of loosely coupled processors. An object-based programming language encourages a methodology for designing and creating a program as a set of autonomous components, whereas a distributed operating system permits a collection of workstations or personal computers ...

Keywords: capability scheme, distributed operating systems, error recovery, method invocation, nested transaction, object model, object reliability, object-based programming languages, processor allocation, resource management, synchronization, transaction

6 Transactions and synchronization in a distributed operating system

Matthew J. Weinstein, Thomas W. Page, Brian K. Livezey, Gerald J. Popek
December 1985 **ACM SIGOPS Operating Systems Review , Proceedings of the tenth ACM symposium on Operating systems principles SOSP '85**, Volume 19 Issue 5
Publisher: ACM Press

Full text available: pdf(974.32 KB) Additional Information: full citation, references, citings, index terms

7 Minimum cost adaptive synchronization: experiments with the ParaSol system

Edward Mascarenhas, Felipe Knop, Reuben Pasquini, Vernon Rego
October 1998 **ACM Transactions on Modeling and Computer Simulation (TOMACS)**, Volume 8 Issue 4
Publisher: ACM Press

Full text available: pdf(265.07 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a novel adaptive synchronization algorithm, called the minimum average cost (MAC) algorithm, in the context of the parasol parallel simulation system. ParaSol is a multithreaded system for parallel simulation on shared- and distributed-memory environments, designed to support domain-specific Simulation Object Libraries. The proposed MAC algorithm is based on minimizing the cost of synchronization delay and rollback at a process, whenever its simulation driver must decide whether ...

Keywords: ParaSol, adaptive synchronization, optimal delay, optimistic synchronization, parallel and distributed simulation, stochastic simulation, thread

8 Principles and realization strategies of multilevel transaction management

Gerhard Weikum
March 1991 **ACM Transactions on Database Systems (TODS)**, Volume 16 Issue 1
Publisher: ACM Press

Full text available: pdf(3.72 MB) Additional Information: full citation, abstract, references, citings, index terms, review

One of the demands of database system transaction management is to achieve a high

degree of concurrency by taking into consideration the semantics of high-level operations. On the other hand, the implementation of such operations must pay attention to conflicts on the storage representation levels below. To meet these requirements in a layered architecture, we propose a multilevel transaction management utilizing layer-specific semantics. Based on the theoretical notion of multilevel serial ...

Keywords: atomicity persistence concurrency control, multilevel transactions, persistence, serializability

9 Preliminary thoughts on problem-oriented shared memory: a decentralized approach to distributed systems

David R. Cheriton

October 1985 **ACM SIGOPS Operating Systems Review**, Volume 19 Issue 4

Publisher: ACM Press

Full text available: pdf(1.05 MB) Additional Information: full citation, abstract, references, citings

Much of the work to date on distributed systems has focused on the correct choice of *communication paradigm*, stressing (for example) message primitives, remote procedure call, problem- oriented protocols and so on. A distributed system service is then implemented as a module executing on particular server machine that is accessed using these communication facilities. In contrast, the shared *memory paradigm* has been used on multiprocessor and uniprocessor systems. In the shared memo ...

10 Concurrency control in advanced database applications

Naser S. Barghouti, Gail E. Kaiser

September 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 3

Publisher: ACM Press

Full text available: pdf(4.69 MB) Additional Information: full citation, references, citings, index terms

Keywords: advanced database applications, concurrency control, cooperative transactions, design environments, extended transaction models, long transactions, object-oriented databases, relaxing serializability

11 Synchronization and recovery in a client-server storage system

E. Panagos, A. Biliris

August 1997 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 6 Issue 3

Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(205.25 KB) Additional Information: full citation, abstract, citings, index terms

Client-server object-oriented database management systems differ significantly from traditional centralized systems in terms of their architecture and the applications they target. In this paper, we present the client-server architecture of the EOS storage manager and we describe the concurrency control and recovery mechanisms it employs. EOS offers a semi-optimistic locking scheme based on the multi-granularity two-version two-phase locking protocol. Under this scheme, multiple concurrent reade ...

Keywords: Checkpoint, Client-server architecture, Object management, Concurrency control, Locking, Logging, Recovery, Transaction management

12 Distributed transactions for reliable systems

Alfred Z. Spector, Dean Daniels, Daniel Duchamp, Jeffrey L. Eppinger, Randy Pausch

December 1985 **ACM SIGOPS Operating Systems Review , Proceedings of the tenth ACM symposium on Operating systems principles SOSP '85**, Volume 19 Issue 5

Publisher: ACM Press

Full text available: pdf(1.44 MB) Additional Information: full citation, references, citings, index terms

13 Report on the fourth ACM SIGOPS European workshop fault tolerance support in distributed systems

Özalp Babaoğlu

January 1991 **ACM SIGOPS Operating Systems Review**, Volume 25 Issue 1

Publisher: ACM Press

Full text available: pdf(1.76 MB) Additional Information: full citation, index terms

14 A VLIW architecture for a trace scheduling compiler

Robert P. Colwell, Robert P. Nix, John J. O'Donnell, David B. Papworth, Paul K. Rodman

October 1987 **ACM SIGARCH Computer Architecture News , ACM SIGPLAN Notices , ACM SIGOPS Operating Systems Review , Proceedings of the second international conference on Architectual support for programming languages and operating systems ASPLOS-II**, Volume 15 , 22 , 21 Issue 5 , 10 , 4

Publisher: IEEE Computer Society Press, ACM Press

Full text available: pdf(1.59 MB) Additional Information: full citation, abstract, references, citings, index terms

Very Long Instruction Word (VLIW) architectures were promised to deliver far more than the factor of two or three that current architectures achieve from overlapped execution. Using a new type of compiler which compacts ordinary sequential code into long instruction words, a VLIW machine was expected to provide from ten to thirty times the performance of a more conventional machine built of the same implementation technology.Multiflow Computer, Inc., has now built a VLIW called the TRACE™< ...

15 Object orientation in multidatabase systems

Evaggelia Pitoura, Omran Bukhres, Ahmed Elmagarmid

June 1995 **ACM Computing Surveys (CSUR)**, Volume 27 Issue 2

Publisher: ACM Press

Full text available: pdf(4.85 MB) Additional Information: full citation, abstract, references, citings, index terms, review

A multidatabase system (MDBS) is a confederation of preexisting distributed, heterogeneous, and autonomous database systems. There has been a recent proliferation of research suggesting the application of object-oriented techniques to facilitate the complex task of designing and implementing MDBSs. Although this approach seems promising, the lack of a general framework impedes any further development. The goal of this paper is to provide a concrete analysis and categorization of the various ...

Keywords: distributed objects, federated databases, integration, multidatabases, views

16 Fast detection of communication patterns in distributed executions

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Publisher: IBM Press

Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

17 Special issue: AI in engineering

D. Sriram, R. Joobbani

April 1985 **ACM SIGART Bulletin**, Issue 92

Publisher: ACM Press

Full text available: pdf(8.79 MB) Additional Information: full citation, abstract

The papers in this special issue were compiled from responses to the announcement in the July 1984 issue of the SIGART newsletter and notices posted over the ARPAnet. The interest being shown in this area is reflected in the sixty papers received from over six countries. About half the papers were received over the computer network.

18 The MARUTI hard real-time operating system

S. Levi, S. K. Tripathi, S. D. Carson, A. K. Agrawala

July 1989 **ACM SIGOPS Operating Systems Review**, Volume 23 Issue 3

Publisher: ACM Press

Full text available: pdf(1.07 MB) Additional Information: full citation, abstract, citings, index terms

The MARUTI operating system is designed to support real-time applications on a variety of hardware systems. The kernel supports objects as primitive entities, and provides a communication mechanism that allows transparent distribution in networked systems. Fault tolerance is provided through replication and consistency-control mechanisms. Most importantly, MARUTI supports guaranteed-service scheduling, in which jobs that are accepted by the system are verified to satisfy general time constraints ...

19 User Recovery and Reversal in Interactive Systems

James E. Archer, Richard Conway, Fred B. Schneider

January 1984 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 6 Issue 1

Publisher: ACM Press

Full text available: pdf(1.30 MB) Additional Information: full citation, references, citings, index terms

20 Special issue on prototypes of deductive database systems: The aditi deductive database system

Jayen Vaghani, Kotagiri Ramamohanarao, David B. Kemp, Zoltan Somogyi, Peter J. Stuckey, Tim S. Leask, James Harland

April 1994 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 3 Issue 2

Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(2.67 MB) Additional Information: full citation, abstract, references, citings

Deductive databases generalize relational databases by providing support for recursive views and non-atomic data. Aditi is a deductive system based on the client-server model; it is inherently multi-user and capable of exploiting parallelism on shared-memory multiprocessors. The back-end uses relational technology for efficiency in the management of disk-based data and uses optimization algorithms especially developed for the bottom-up evaluation of logical queries involving recursion. The front ...

Keywords: implementation, logic, multi-user, parallelism, relational database

Google Scholar BETA

log synchronize OR undoable "file system " Search

Advanced Scholar Search
Scholar Preferences
Scholar Help

Scholar Results **1 - 10** of about **1,800** for **log synchronize OR undoable "file system "**. (**0.10** seconds)

An implementation of a **log**-structured **file system** for UNIX
MI Seltzer, K Bostic, MK McKusick, C Staelin - USENIX Winter, 1993 - citeseer.csail.mit.edu
... An Implementation of a **Log**-Structured **File System** for UNIX An Implementation of a **Log**- ... 201 Page 2. An Implementation of a **Log**-Structured **File System** for UNIX ...
Cited by 142 - View as HTML - Web Search - cse.iitb.ac.in - www-uxsup.csx.cam.ac.uk - pha.com.au - all 28 versions »

DEcorum **File System** Architectural Overview
ML Kazar, BW Leverett, OT Anderson, V Apostolides, ... - Atlanta, 1986 - tedanderson.home.mindspring.com
... be written to disk until the **log** has been ... of the glue layer is to **synchronize** actions on ... operation provided by a conven- tional **file system**, a corresponding ...
Cited by 64 - View as HTML - Web Search - cag.lcs.mit.edu - tedanderson.home.mindspring.com - cag.lcs.mit.edu

The Zebra striped network **file system**
JH Hartman, JK Ousterhout, PR Katz, PR Larson, F ... - 1992 - cl.cam.ac.uk
... I call this form of striping **log**-based striping, and its operation is similar to that of a **log**-structured **file system** (LFS) [Rosenblum91]. ...
Cited by 232 - View as HTML - Web Search - cs.arizona.edu - oslab.snu.ac.kr - ipd.uka.de - all 32 versions »

Overview of the Vesta parallel **file system**
PF Corbett, SJ Baylor, DG Feitelson - ACM SIGARCH Computer Architecture News, 1993 - portal.acm.org
... Users **log** onto the host nodes and use them to acquire user ... export of files between Vesta and an external **file system** is briefly ... I get and / **synchronize** _ __ ...
Cited by 64 - Web Search - portal.acm.org

The Google **File System**
S Ghemawat, H Gobioff, ST Leung - Proceedings of the 19th ACM Symposium on Operating Systems ..., 2003 - portal.acm.org
... The master recovers its **file system** state by replaying the operation **log**. To minimize startup time, we must keep the **log** small. ...
Cited by 75 - Web Search - csie.fju.edu.tw - cs.cornell.edu - ppgia.pucpr.br - all 74 versions »

Taming aggressive replication in the Pangaea wide-area **file system**
Y Saito, C Karamanolis, M Karlsson, M Mahalingam - ACM SIGOPS Operating Systems Review, 2002 - portal.acm.org
... consistency, that use locks, or that **synchronize** using directory ... 22] is a peer-to-peer **file system** that lets ... Its **log**-based update propagation also allows for ...
Cited by 60 - Web Search - hpl.hp.com - ececs.uc.edu - cse.nd.edu - all 15 versions »

The Alpine **file system**
MR Brown, KN Kolling, EA Taft - ACM Transactions on Computer Systems, 1985 - portal.acm.org
... The **log**-based **file system** can be structured so that most of it does not depend on details of the underlying **file system** and can be easily ported from one **file** ...
Cited by 33 - Web Search - portal.acm.org - csa.com - Library Search

The Swarm Scalable Storage System
JH Hartman, I Murdock, T Spalink - PROC INT CONF DISTRIB COMPUT SYST., 1999 - ieeexplore.ieee.org
... example, the reader of a file must **synchronize** with the writer, even if the underlying **file system** provides strong ... **Log** layer 2.1.1. **Log** format The **log** is a ...
Cited by 42 - Web Search - doi.ieeecs.org - doi.ieeecomputersociety.org - cs.arizona.edu - all 8 versions »

A caching **file system** for a programmer's workstation
MD Schroeder, DK Gifford, RM Needham - ACM SIGOPS Operating Systems Review, 1985 - portal.acm.org
... special purposes such as updating local **log** files ... The Swallow **file system** design [13] first explored the ... provided within a workstation to **synchronize** such local ...
Cited by 55 - Web Search - cs.biu.ac.il - portal.acm.org - Library Search

The Global **File System**
S SOLTIS, T RUWART, M OKEEFE - Minnesota Univ, Fifth NASA Goddard Conference on Mass ..., 1996 - parl.clemson.edu
... client can reconnect to the network and **synchronize** its cache ... The xFS **file system** is part of the Berkeley's ... It uses a **log** structured approach like Sprite's ...
Cited by 39 - View as HTML - Web Search - csa.com

Gooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

log synchronize OR undoable "file s) Search

Google Home - About Google - About Google Scholar